Motion Imagery Standards Board Draft Recommended Practice Common Time Reference for Digital Motion Imagery Using Coordinated Universal Time (UTC)	MISB RP 0603 Date 10 August 2006

1 Scope

Digital motion imagery is employed by the Department of Defense (DoD) / Intelligence Community (IC) for a wide range of applications throughout the world and is correlated with information from many sources including other motion imagery systems, other sensor systems, other intelligence systems, and DoD services' operations. Time is one critical reference to correlate information from these systems and operations. The DoD / IC community uses Coordinated Universal Time (UTC) as a common time reference. This document defines the Recommended Practice (RP) for setting and using common UTC time reference for digital motion imagery.

There are two objectives for implementing a deterministic common time reference in digital motion imagery systems.

1) Correlation of digital motion imagery frames with the sensor / platform metadata to produce metadata associated with each motion imagery frame selected. Examples include: image center and other metadata for frames for situational awareness and precision metadata for frames to perform photogrammetric analysis and targeting mission support.

2) Interoperability and exchange of common time referenced motion imagery and motion imagery products with other sensor systems, other collection systems, other intelligence systems, and the warfighter. Examples include: cooperative motion imagery sensors / platforms supporting a single mission objective, and multi-INT exploitation of digital motion imagery with other intelligence or information sources.

This RP defines the use of UTC as the deterministic common time reference for correlation of motion imagery frames and metadata and to facilitate interoperability among DoD / IC systems and other systems.

This RP defines of the use of the US Global Positioning System (GPS) as the precision time reference source for calculating UTC using a normative mathematical transformation defined in this RP.

This RP identifies two time formats in common use in digital motion imagery systems today: Society of Motion Picture and Television Engineers (SMPTE) 12M time formats and microseconds since 1970 time format. SMPTE 12M is used in commercial video products to identify each frame in a file or stream. Microseconds since 1970 is deried from the POSIX standard and is used in computer based motion imagery applications.

This RP provides informative mathematical relationships to reformat UTC into SMPTE 12M time format and microseconds since 1970 time format.

This RP is designed to be the parent document of a series of documents outlining MISB RP's and Engineering Guidelines (EG's) for time stamping compressed, uncompressed and other motion imagery formats as outlined in Figure 1.

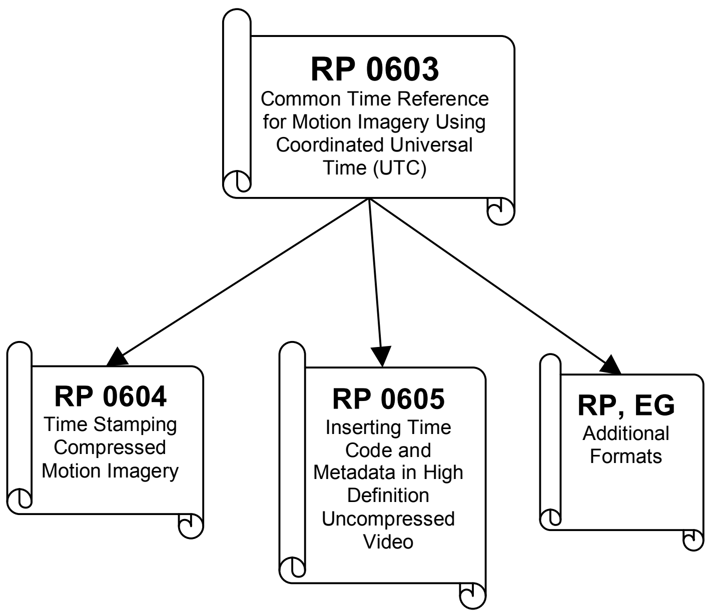


Figure 1, Time Stamping Document Structure

2 References

The following documents are provided as normative references to this guideline:

MISP STANDARD 9708 - Imbedded Time Reference for Motion Imagery Systems.

MISP STANDARD 9715 - Time Reference Synchronization.

SMPTE 12M-1999 – Time and Control Code.

SMPTE 309M-1999 – Transmission of Date and Time Zone Information in Binary Groups of Time Control Code.

SMTPE EG40-2002 – Conversion of Time Values Between SMPTE 12M Time Code, MPEG-2 Program Clock Reference (PCR) Time Base and Absolute Time.

Assistant Secretary of Defense for Command, Control, Communications and Intelligence, “Global Positioning Standard Positioning Service Performance Standard”, Sections 1.4, 1.4.2, A-1.3.2.3, A2.4

MISP RP 0604 – Time Stamping Compressed Motion Imagery

MISP RP 0605 - Inserting Time Code and Metadata in High Definition Uncompressed Video

3 Introduction

This RP identifies time stamping objectives and practices for real-time insertion of precision time stamps into streaming motion imagery and associated metadata. Technical implementation details for each system implementation are within the system engineering “trade space” for each system and each system’s mission requirements. These implementation decisions will drive the system complexity, performance, cost and size / weight / power. Specific implementation decisions will determine the accuracy and precision with which motion imagery and associated metadata can used to meet mission requirements such as warfighter situational awareness, intelligence imagery exploitation, operational targeting missions, library services, etc.

The following figure illustrates the relationships of time references and formats defined in this RP.

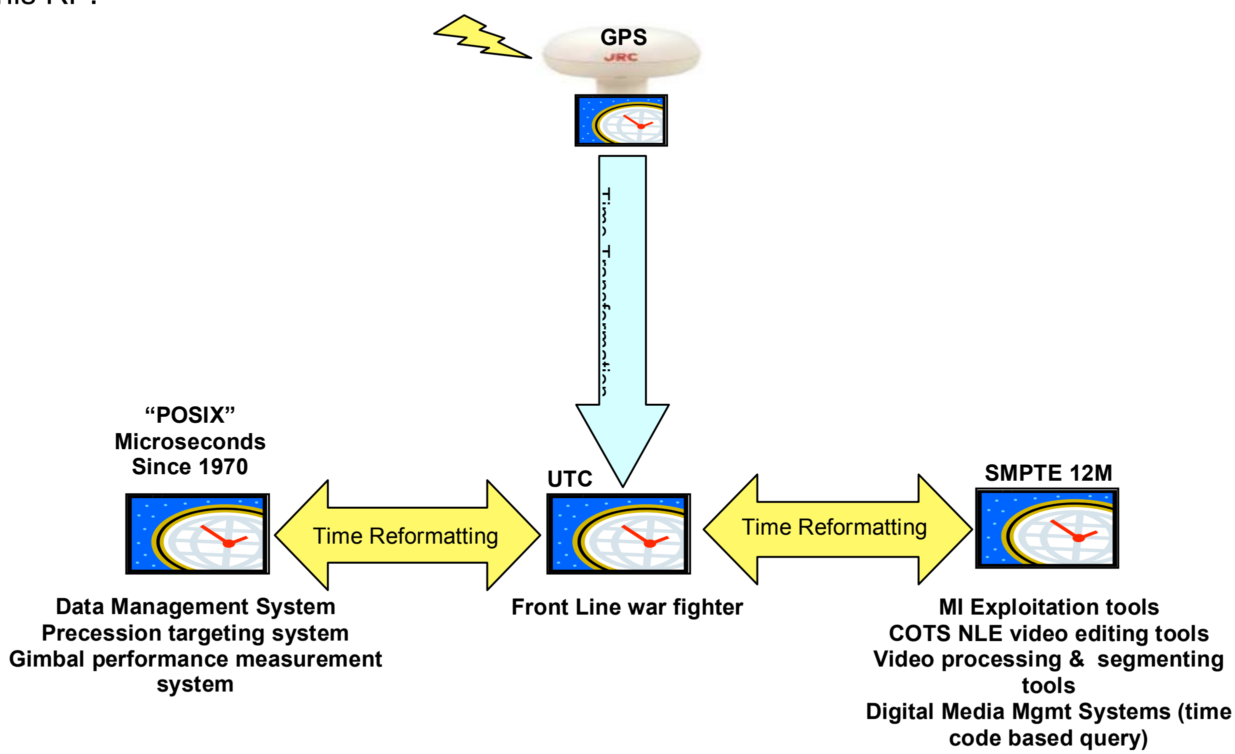


Figure 2, Time Type Use Cases

UTC is the normative time reference for motion imagery systems required by this RP. UTC is a high-precision atomic time standard. It is the basis for military and legal civil time all over the Earth. It is also referred to as Zulu time (Z), or as the equivalent "Greenwich Mean Time" (GMT). UTC time reference can be obtained from a number of alternate sources, however using the Global Positioning System’s (GPS) operated by the United States is the recommended time reference source for this RP. GPS was synchronized to UTC in 1980, and is kept in close synchronization with International Atomic Time (TAI). However, due to changes in the earths’ orbit and even adoption of changes in the duration of the TAI second, GPS time reference is no longer exactly the same as UTC time reference, differing by discrete time offsets known as “leap seconds”, which are updated periodically by US and international standards bodies.

This RP defines the normative mathematical transformation from GPS time reference to UTC time reference, which is defined in Section 4, “UTC from GPS as an Absolute Timing Reference for Motion Imagery”.

Annex A of this RP informatively provides methods by which UTC can be reformatted into SMPTE time formats, used for video systems, and “microseconds since 1970”, often used as a precision time format for metadata capture, event capture, and precise metadata calculations. The mathematical relationships used to reformat UTC into SMPTE 12M time formats and “microseconds since 1970” time format are defined in Annex A, “Time Conversion Formulas (Informative)”.

Methods describing the time stamping of the metadata and video frames are defined in this RP. These methods ensure a common and consistent interpretation of events.

4 UTC from GPS as the Common Time Reference for Motion Imagery

GPS provides position and time information enabling geolocation anywhere on the earth. GPS system consists of a constellation of 24 satellites, plus spares, orbiting the earth twice each day at an altitude of 12,500 miles. Each satellite transmits frequency, time and date information to airborne and ground based GPS receivers. The use of UTC as derived from GPS as the common time reference for time stamping motion imagery enables the frame-accurate temporal fusion of motion imagery streams from multiple sensors located worldwide.

4.1 Common Time Reference

Correlation of temporal events acquired from motion imagery sensors is critical for monitoring the nature of activities in the field of view as they change continuously over time. Post event analysis is enhanced with temporal synchronization of multiple video streams and associated metadata. Temporal motion imagery and metadata synchronization is valuable for utilizing multiple sensors to synchronously “re-play” motion imagery and metadata segments from all sensors involved in a given mission. This example application has value for collections when actions of interest are viewable from sensors collecting in a wide field of view mode, but not viewable from sensors collecting on the same event in spotter mode where the action of interest is outside of view. The implementation of a common time reference enables frame-accurate synchronization of multiple fields of view and metadata from multiple sensors. This common time reference enables temporal fusion for post-mission analysis.

UTC, derived from GPS time, maintains synchronization with the official time kept by the U.S. Naval Observatory's Master Clock to within one millisecond. Today's motion imagery sensors commonly operate at either 30 or 60 frames per second (FPS), resulting in a frame interval in the 33 to 17 ms range. Therefore, global synchronization of shutters and time code can be obtained at sub-frame accuracy.

4.1.1 Transformation of GPS Time to UTC

Many GPS receivers output the UTC time, plus a one pulse per second (1-PPS) synchronization signal; others output the 1-PPS but report GPS time and UTC offset separately requiring external calculation of UTC = GPS - Leap Seconds. Leap seconds are often provided by the GPS receiver. Some GPS receivers may output Inter-Range Instrumentation Group (IRIG) time formats which provide equivalent synchronization and time information.

A few low cost receivers only provide GPS Week and GPS Seconds parameters, leaving the entire time calculation to external devices. The offset of GPS Seconds is defined as the beginning of the current GPS week. GPS time is referenced to a UTC zero time point originally defined as midnight (00:00 UTC) before the morning of 1980-01-06. However, as the GPS Week parameter is only 10 bits, weeks are numbered from 0 to 1023, then roll back to 0 and are again numbered from zero up. The GPS week cycle is 1024 weeks (7168 days, or 19+ years); the latest zero point was 1999-08-22 00:00 GPS time. The following allows calculation of date and time to one second (further precision may require provisions such as a local oscillator synchronized to the GPS signal):

```
If (GPS Seconds is less than Leap Seconds)
    GPS Seconds = GPS Seconds + 604800
    GPS Week = GPS Week - 1
End If
Beginning_of_current_week = (7 × GPS Week) + 1999-08-22 00:00
Day_of_week = (GPS Seconds - Leap Seconds) ÷ 86400
Current_date = Beginning_of_current_week + Day_of_week
Seconds_from_midnight = (GPS Seconds - Leap Seconds) % 86400
Hour = Seconds_from_midnight ÷ 3600
Minute = (Seconds_from_midnight % 3600) ÷ 60
Second = Seconds_from_midnight % 60
```

where: × is multiplication
÷ is integer division without rounding
% is the modulus operator (remainder after integer division).
Note that this increases the range of the "GPS Week" and "GPS Seconds"

variables from the range provided from the GPS device (as explained in the preceding paragraph).

Annex A of this document provides an informative definition on how to convert UTC time to the applicable clock formats common to most sensors (analog and digital).

5 Motion Imagery Time Synchronization Methods

Motion imagery systems can be implemented with varying methods of synchronization based upon system requirements.

The methods of synchronization depend on when, where, and how the video, metadata and additional supporting information are time stamped and correlated. This section defines time stamping video and metadata only.

5.1 Video and Metadata Time Synchronization

Motion imagery system designers (and users of their systems) must be aware of how accurate (time error) the timestamp is for each metadata element. Designers shall provide metadata defining the time stamping method implemented for video and metadata in the system design to include when image exposure occurred and time stamped. This enables inclusion of this information within the metadata package. As a general rule, metadata elements should include all necessary information to precisely determine timing of an event. Motion imagery analysts may require precise time-stamped metadata for analysis tasks that depend on the correlation of data to video frames.

5.2 Metadata Time Synchronization Methods

Metadata is extracted from the sensor system, aircraft systems, and / or ground systems and is time stamped as a metadata item or a package of metadata. Metadata time synchronization is best described by three traits: tagging, sampling, and synchronization. The characteristics of each trait are listed from least accurate to most accurate.

5.2.1 Tagging

Non-Time-Tagged – Metadata items that do not have an associated time tag.

Time-Tagged – Metadata items that have an associated time tag. This is the preferred approach.

5.2.2 Sampling

Buffered – Metadata items that are created at various sample rates and written into a buffer. The buffer is then periodically output as a metadata message. If applying time tags then the metadata message is time-tagged when it is output.

Filtered – Metadata items that are collected over some time period and interpolated to create a new sample at an intermediate time. If applying time tags the interpolated metadata message is time tagged with the intermediate time tag.

Event – Metadata items that are created based on an event. The items are either output individually or buffered into a group and output as a metadata message.. If applying time tags then the event is time tagged with the event time.

Individual – Metadata items that are created at various independent sample rates. If applying time tags then the individual items are time tagged with their creation time. The items are either output individually or buffered into a group and output as a metadata message.

Group – Metadata items that are created at the same sample rate and output as a metadata message. If applying time tags then the metadata message is time-tagged with the sample time of the group.

5.2.3 Synchronization

Video Asynchronous – Metadata sampling that is unrelated to the video clock or frame time.

Video Isochronous – Metadata items that are sampled at a rate synchronous to the video clock, but not necessarily at the video frame time (i.e. out of phase from the video frame time).

Video Synchronous – Metadata items are sampled at the video frame time.

	NON TIME TAGGED	TIME TAGGED	BUFFERED	FILTERED	EVENT	INDIVIDUAL	GROUP	VIDEO ASYNCHRONOUS	VIDEO ISOCHRONOUS	VIDEO SYNCHRONOUS
Tagging		X								
Sampling							X			
Sync										X

Table 1, Sampling and Time Tagging Methods

As shown in Table 1 the Time-Tagged, Group Sampled, Video Synchronous metadata provide the most accurate method for time stamped metadata.

5.3 Video Time Synchronization Methods

A video frame and associated time stamp is defined to be the start of the actual image capture process, such as the opening of the shutter or start of Charge Coupled Device (CCD) integration.

A further consideration is the source of the synchronization signal, specifically: an external signal source such as a “genlock” signal or an externally derived clock source such as UTC clock signal; or an internally derived sensor / camera clock signal.

Video time stamping can be accomplished using a variety of methods which are described by four traits: tagging, genlock, frame rate, and tag insertion. The characteristic of each trait is listed from least accurate to most accurate.

5.3.1 Tagging

Non-Time-Tagged – Video frames that do not have an associated time tag.

Time-Tagged – Video frames that have an associated time tag.

5.3.2 Genlock

Asynchronous – Sensor clock that is not synchronized to the common time reference.

Synchronous – Sensor clock that is synchronized (genlocked) to the common time reference.

5.3.3 Frame Rate

Non-Integer Frame Rate – Frame rates where number of frames per second is not an integer. This typically includes frame rates of 60/1.001, 30/1.001, or 24/1.001 frames per second.

Integer Frame Rate – Frame rates that have an integer number of frames per second. This typically includes frame rates of 24, 25, 30, 50, or 60 frames per second.

5.3.4 Tag Insertion

Post Processing Tag Insertion – Video frames that are time tagged in a post processing environment. The relationship between this time and the creation time may not be precisely defined, but may provide for reasonable association with the metadata.

Decoding Tag Insertion– Video frames that are time tagged when decoded in intermediate equipment. There is a deterministic delay between this time and the creation time at the sensor.

Acquisition Tag Insertion– Video frames are time tagged at creation at the sensor.

	NON TIME TAGGED	TIME TAGGED	ASYNCHRONOUS	SYNCHRONOUS	NON-INTEGER FRAME RATE	INTEGER FRAME RATE	POST PROCESSING	DECODING	ACQUISITION
Tagging		X							
Genlock				X					
Frame Rate						X			
Tag Insertion									X

Table 2, Video Time Tagging Methods

Using these traits: Time Tagged, Synchronous, Integer Frame Rate, Acquisition Tagged video frames is the most accurate method.

5.4 Frame Rate Importance - Non-Integer (Drop-Frame Counts) versus Integer (Non Drop-Frame Count)

Integer frame rate versus non-integer frame rate is an important consideration for SMPTE 12M time stamped video. While typically used with non-integer frame rates, drop-frame counts may or may not be used in cases with non-integer frame rates. Not using drop-frame counts when using a non-integer frame rate is the very worst case scenario. Using drop-frame counts with non-integer frame rates is better, but still limited to 60-75 millisecond accuracy. Integer frame rates should be used where possible.

6 Glossary of acronyms

C4ISR: Command, Control, Communication, Computers, Intelligence, Surveillance, Reconnaissance
DF: Drop Frame Count
EG: Engineering Guideline
FPS: Frames Per Second
GPS: Global Positioning System
IRIG: Inter-Range Instrumentation Group
LTC: Longitudinal Time Code
MI: Motion Imagery
MISP: Motion Imagery Standards Profile
NDF: Non-Drop Frame Time Code
NLE: Non-linear Editor
UTC: Coordinated Universal Time ("Zulu Time")
VANC: Vertical Ancillary Data
VTR: Video Tape Recorder

Annex A Time Conversion Formulas (Informative)

1 Reformatting of UTC to Microseconds Since 1970

Microseconds since 1970 are identified as a machine readable unsigned 64 bit integer containing microseconds since midnight January 1st 1970. Microseconds since 1970 can be utilized in synchronous systems to efficiently compare video, audio, and metadata time stamps. Note: All computers do not implement *microseconds since 1970-01-01 00:00* identically, so this count must regularly be recalibrated to UTC by adjusting the computer real-time clock.

The following formula can be used to reformat UTC to Microseconds since 1970:

Years = Years – 1970

Days = Day_of_year + Leap_days + 365 × (Years)

Seconds = (86,400 × Days) + (3600 × Hour) + (60 × Minute) + Second

Microseconds = (1,000,000 × Seconds) + (1,000 × milliseconds)

where Leap_days occur in years divisible by 4, except years divisible by 100 but they do occur in years divisible by 400

2 Reformatting of UTC to SMPTE 12M

SMPTE 12M is a frame counting labeling standard originally developed to facilitate frame accurate video editing and synchronizing film sound-tracks. The timing data in SMPTE 12M timecode takes the form of an eight digit twenty-four hour clock. The count consists of 0 to 59 seconds, 0 to 59 minutes and 0 to 23 hours. The second is subdivided into a number of frames, which may be varied to match the various frame-rates used around the world. The frame-rate is the number of times a second that the picture is updated.

Historically, SMPTE 12M timecode has been employed as a relative time reference within a linear frame sequence, such as video tape or a video file, for applications such as editing and having no inherent relationship to external time references such as GPS and UTC or "wall clock". However, current applications of SMPTE 12M, including certain commercial components, do provide for the use of an external time and control code sources, such as a GPS or UTC time reference, to synchronize the SMPTE 12M timecode to external time sources.

Despite the provision for setting SMPTE 12M timecode to an external time reference at collection, such as GPS or UTC, this timecode is often not maintained in derived products when the content is processed through video processing systems, such as non-linear editors.

It is recognized that legacy NTSC based sensor and video processing systems are currently operational and will continue to be operated for some period into the future. These systems use a frame rate of 29.97 (30/1.001) or 60/1.001. Commercial industry employs the SMPTE 12M “drop-frame” (DF) timecode to provide relatively accurate alignment of SMPTE 12M timecode for NTSC video with external time references, such as GPS and UTC or a “wall clock”, to insure that a 60 minute program, measured using SMPTE 12M DF time code, is truly 60 minutes long when broadcast. The term drop frame can be misleading in that the term suggests that a video frame or image is dropped. This is not the case, what is done is frame label count “skips” one or more frame counts to periodically realign the 12M timecode with the “wall clock”.

Without such a scheme, the 12M timecode would be “off” by 108 frames (or +3.6 sec) in one hour of running time. When drop-frame compensation is applied to an NTSC television time code, the total error accumulated after one hour is reduced to 3.6 ms.. For example, in non-drop frame count mode, the counter goes from 01:12:59:29 to 01:13:00:00. These are the SMPTE numbers showing hours, minutes, seconds, and frames. After the 29th frame occurs the clock shows the next second and the zero frame. In the drop frame count mode, the changeover looks like this: 01:12:59:29 to 01:13:00:02. Notice the frame number 00 and 01 are dropped. These occasional dropped frames add up to 108 at the end of an hour, making time code "time" and clock time match exactly. However, the scheme itself is an artificial solution leading to difficult complex implementations.

Since the SMPTE 12M format:

- Cannot provide a precise time below the frame rate (typically 16 and 33 ms resolution)
- Typically does not carry date information required to fully specify the date/time
- Is often not maintained (e.g. it is modified when the imagery is post processed)

SMTPE 12M non-integer drop frame count should not be used in new systems to carry the collection time.

At a minimum, it is required that all new systems that output digital motion imagery intended for frame-accurate exploitation and segment editing produce motion imagery at true integer frame rates (e.g. 2 FPS, 24 FPS, 25 FPS, 30 FPS, 60 FPS, etc) using “non-drop frame count” (NDF) SMPTE 12M time code generated from GPS or UTC as the time reference.

2.1 SMPTE 12M relationship to UTC

For systems that must deal with SMPTE 12M (Both DF and NDF), the following informational algorithms are provided. The mathematical relationships provided below were derived using information presented in SMTPE EG40-2002 – Conversion of Time Values between SMPTE 12M Time Code, MPEG-2 PCR Time Base and Absolute Time.

The UTC time needs to be converted to SMPTE 12M time format to ensure proper interoperability with commercial off the shelf editing, displaying, and exploitation software and hardware.

2.1.1 UTC to SMPTE 12M (integer based frame rates)

The following formula can be used to convert UTC time to SMPTE 12M time format when the video frame_rate, in terms of frames per second, is an integer based number (Non-Drop frame count):

Note: SMPTE 12M is a *time* format without a *date* component. The following formula can be used to convert UTC time to SMPTE 12M time format when the video frame rate is an integer based number (Non-Drop frame count), but the date is ignored:

$$frame\_count = \mathrm{int}(frame\_rate \times time)$$

$$hours = \mathrm{int}\left(\frac{frame\_count}{frame\_rate \times 60 \times 60}\right)$$

$$minutes = \mathrm{int}\left(\frac{frame\_count - frame\_rate \times 60 \times 60 \times hours}{frame\_rate \times 60}\right)$$

$$seconds = \mathrm{int}\left(\frac{frame\_count - frame\_rate \times 60 \times (minutes + 60 \times hours)}{frame\_rate}\right)$$

$$frames = frame\_count - frame\_rate \times (seconds + 60 \times (minutes + 60 \times hours))$$

where: $\times$ is multiplication
Int () is the integer portion of the result within the parenthesis.
Time is defined as the number of seconds since midnight as a rational number.
The result of the divisions are real numbers as opposed to integers.

2.1.2 UTC to SMPTE 12M for non-integer based frame rates

The following formula can be used to convert UTC time to SMPTE 12M time format when the video frame rate is a non-integer based number (Drop frame count):

Note: SMPTE 12M is a *time* format without a *date* component. The following formula can be used to convert UTC time to SMPTE 12M time format when the video frame rate is a non-integer based number (Drop frame count), but the date is ignored:

$$frames\_per\_hour = frame\_rate \times 60 \times 60 - dropped\_frames\_per\_hour$$
$$frame\_count = \mathrm{int}(frame\_rate \times time)$$

$$hours = \operatorname{int}\left(\frac{frame\_count}{frames\_per\_hour}\right)$$

$$minutes = \operatorname{int}\left(\frac{1}{60 \times frame\_rate} \times \left(\begin{array}{l} frame\_count \\ + 2 \times \operatorname{int}\left(\dfrac{frame\_count - (frames\_per\_hour \times hours)}{60 \times frame\_rate}\right) \\ - 2 \times \operatorname{int}\left(\dfrac{frame\_count - (frames\_per\_hour \times hours)}{600 \times frame\_rate}\right) \\ - frames\_per\_hour \times hours \end{array}\right)\right)$$

$$seconds = \operatorname{int}\left(\frac{1}{frame\_rate} \times \left(\begin{array}{l} frame\_count \\ - (frame\_rate \times 60 - 2) \times minutes \\ - 2 \times \operatorname{int}\left(\dfrac{minutes}{10}\right) \\ - frames\_per\_hour \times hours \end{array}\right)\right)$$

$$frames = frame\_count - frame\_rate \times (seconds + 60 \times (minutes + 60 \times hours)) \\ +2 \times minutes - 2 \times \operatorname{int}\left(\frac{minutes}{10}\right) + 180 \times hours$$

where: × is multiplication
Int () is the integer portion of the result within the parenthesis.
108 is the number of *dropped_frames_per_hour* for a 29.97 *frame_rate.*
Time is defined as the number of seconds since midnight as a rational number.
The result of the divisions are real numbers as opposed to integers.

2.1.3 SMPTE 12M to UTC for integer based frame rates

Note: SMPTE 12M is a *time* format without a *date* component. The following formula can be used to convert SMPTE 12M to UTC time when the video frame rate is an integer based number (Non-Drop frame count), but the date must be separately provided:

$$time_{seconds} = \left(\frac{1}{frame\_rate}\right) \times \left(\begin{array}{l} frames \\ + \ frame\_rate \times \left(\begin{array}{l} seconds \\ + \ 60 \times \left(\begin{array}{l} minutes \\ + \ 60 \times hours \end{array}\right) \end{array}\right) \end{array}\right)$$

where: $\times$ is multiplication
Time is defined as the number of seconds since midnight as a rational number.
The result of the divisions are real numbers as opposed to integers.

2.1.4 SMPTE 12M to UTC for non-integer based frame rates

Note: SMPTE 12M is a *time* format without a *date* component. The following algorithm can be used to convert SMPTE 12M to UTC time, when the video frame rate is a non-integer based number (Drop frame count), with a maximum error of 61 milliseconds:

$$frames\_per\_hour = frame\_rate \times 60 \times 60 - dropped\_frames\_per\_hour$$

$$time_{seconds} = \left(\frac{1}{frame\_rate}\right) \times \left(\begin{array}{l} frames \\ + frame\_rate \times (seconds + 60 \times (minutes + 60 \times hours)) \\ - 2 \times minutes \\ + 2 \times \mathrm{int}\left(\frac{minutes}{10}\right) \\ - dropped\_frames\_per\_hour \times hours \end{array}\right)$$

where: × is multiplication

108 is the number of *dropped_frames_per_hour* for a 29.97 *frame_rate*

Time is defined as the number of seconds since midnight as a rational number.

The result of the divisions are real numbers as opposed to integers.

Annex B Metadata Timing Examples (Informative)

1 Metadata Timing Sequence

The following diagram illustrates the timing sequence of both the video frames and some metadata; this diagram will be used in the latter sections to describe the various time tagging techniques. Each row of metadata has different frequency rate of measure (or computation) which may or may not be aligned with the frame rate (usually not). This diagram does not show the latency of each metadata item; some items may take a ½ second to measure so the value reported is a ½ second behind in latency. Furthermore the diagram illustrates all of the measures at a regular rate, which may not actually be the case. Some examples will use different rates of sample than illustrated here.

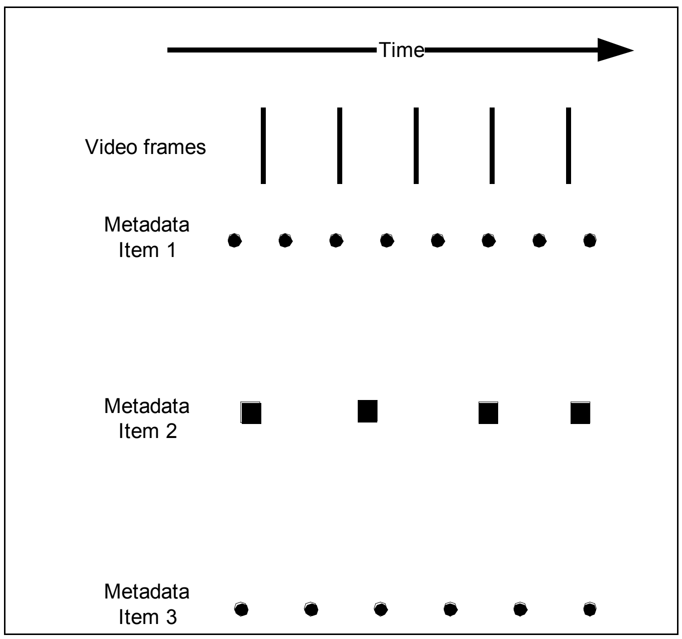


Figure 3, Metadata Timing Sequence

1.1 Non-Time Tagged, Buffered, Video Asynchronous Metadata

Buffered metadata reported by the sensor with no correlation or reference to the time it was created. The metadata is collected by the buffer then output as a metadata message. This is the least accurate method of metadata time synchronization.

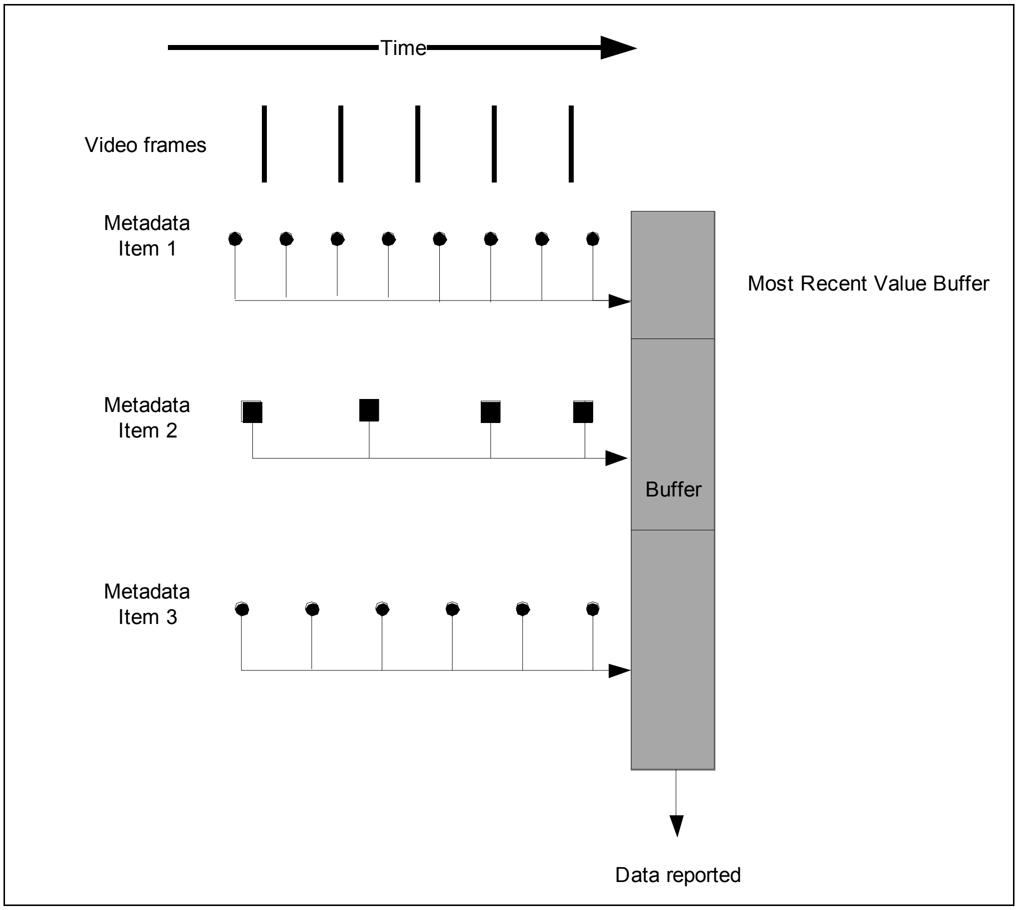


Figure 4, Non-Time Tagged, Buffered, Video Asynchronous Metadata

1.2 Non-Time Tagged, Individual, Video Asynchronous Metadata

The following diagram illustrates individual metadata items sampled without time tagging.

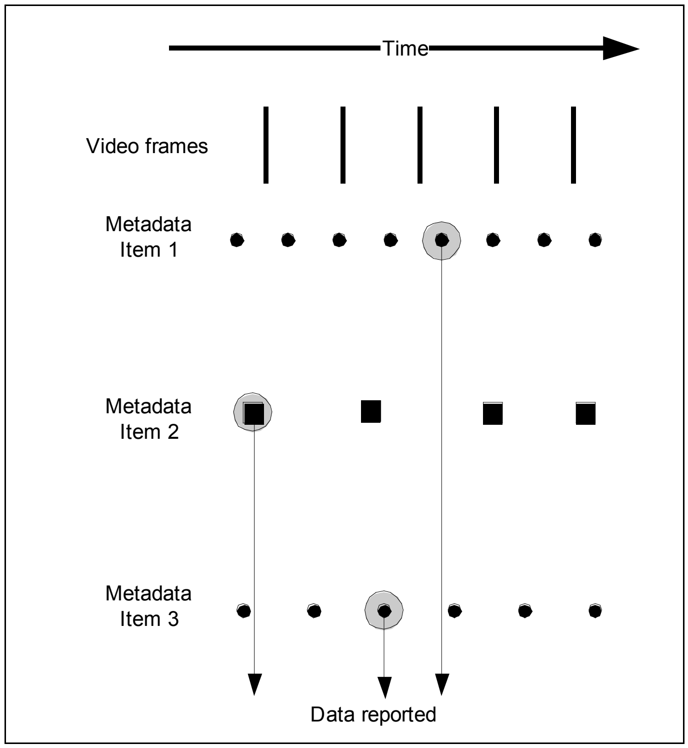


Figure 5, Non-Time Tagged, Individual, Video Asynchronous Metadata

1.3 Time Tagged, Buffered, Video Asynchronous Metadata

Buffered metadata is reported by the sensor with correlation to the time that the buffer is output. The metadata is collected by the buffer then output as a metadata message.

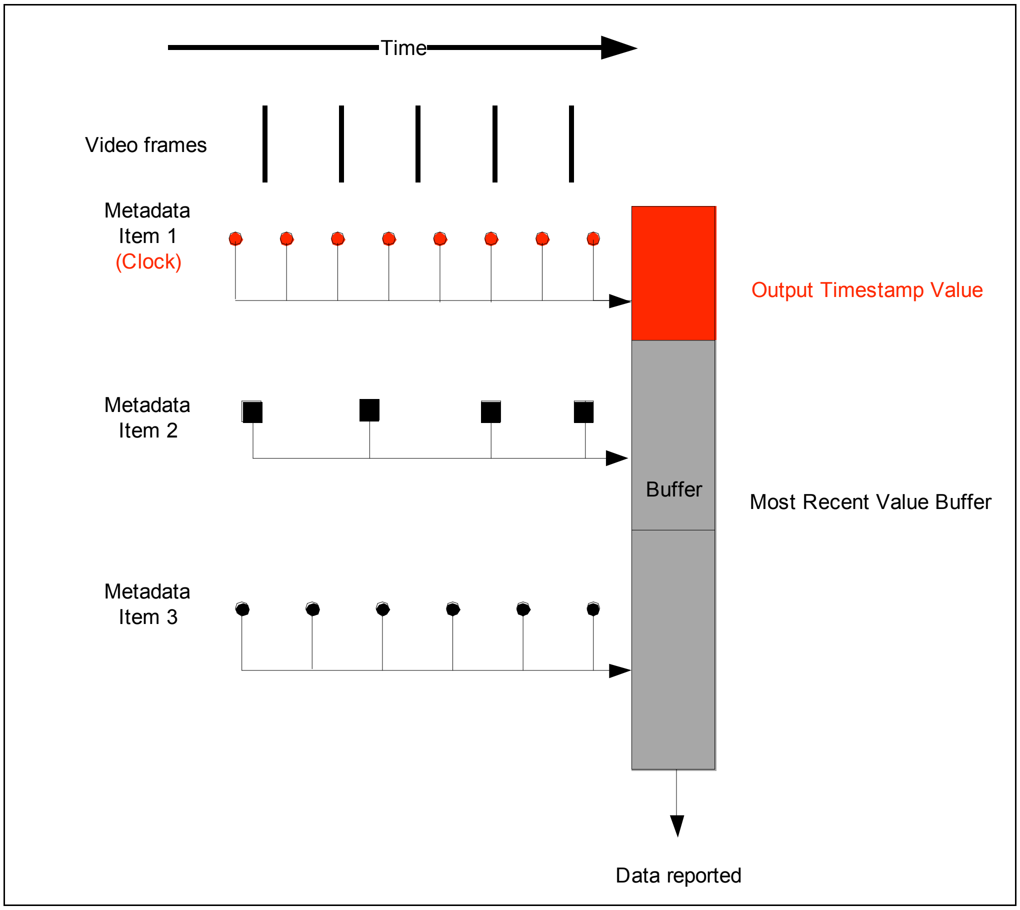


Figure 6, Time Tagged, Buffered, Video Asynchronous Metadata

1.4 *Time Tagged, Filter Sampled, Video Synchronous Metadata*

Filters are used to interpolate the metadata samples. The Time filter reports the intermediate time of the interpolated values. If the intermediate time is video synchronous then the intermediate time stamp will be aligned with the video frames; as shown in the diagram below.

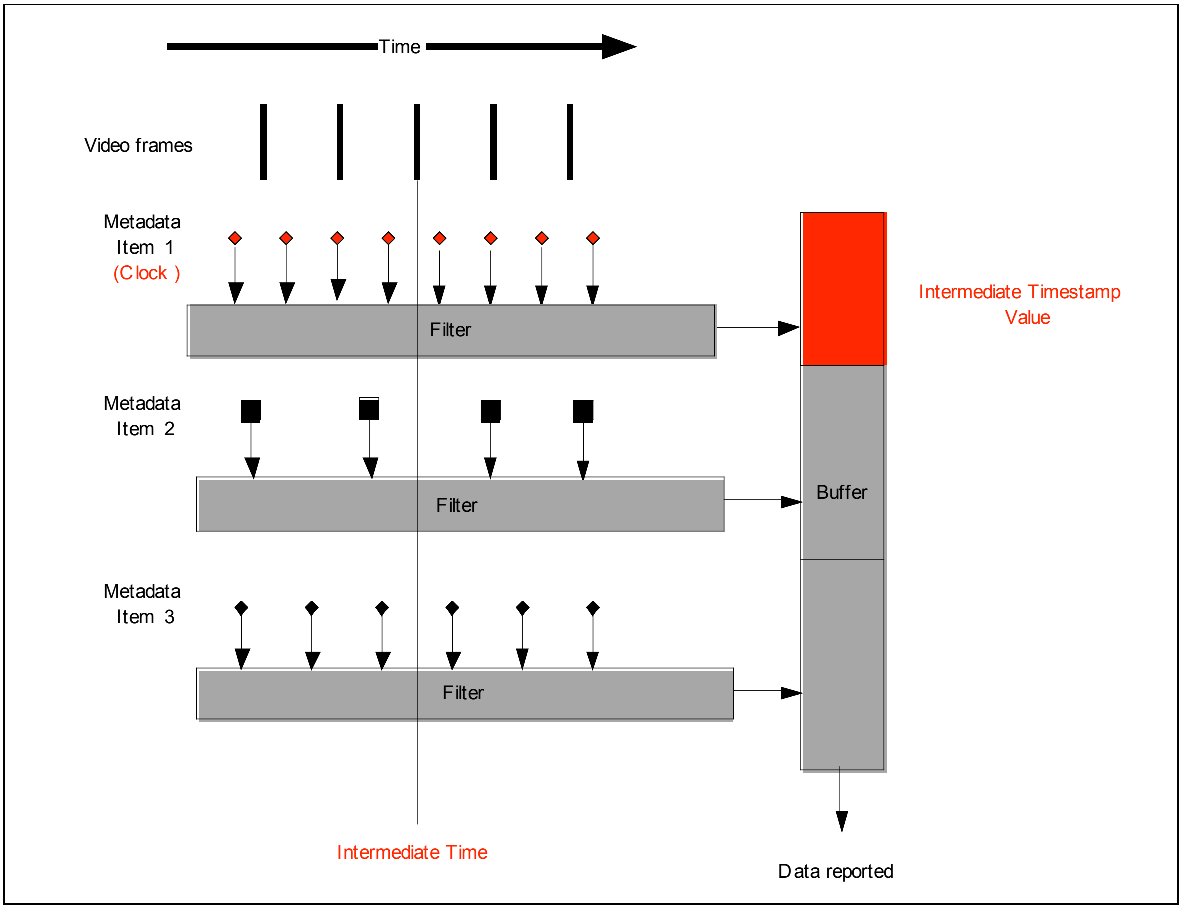


Figure 7, Time Tagged, Filter Sampled, Video Synchronous Metadata

1.5 Time Tagged, Event Sampled, Video Asynchronous Metadata

This is an example of an event driven metadata that is created and time stamped when a predetermined event occurs. An example would be the firing of a Laser Range Finder (LRF); the metadata is gathered from the LRF (and possibly other sources) and it is time stamped at time of triggering. This demonstrates that the metadata items are measured at the time requested.

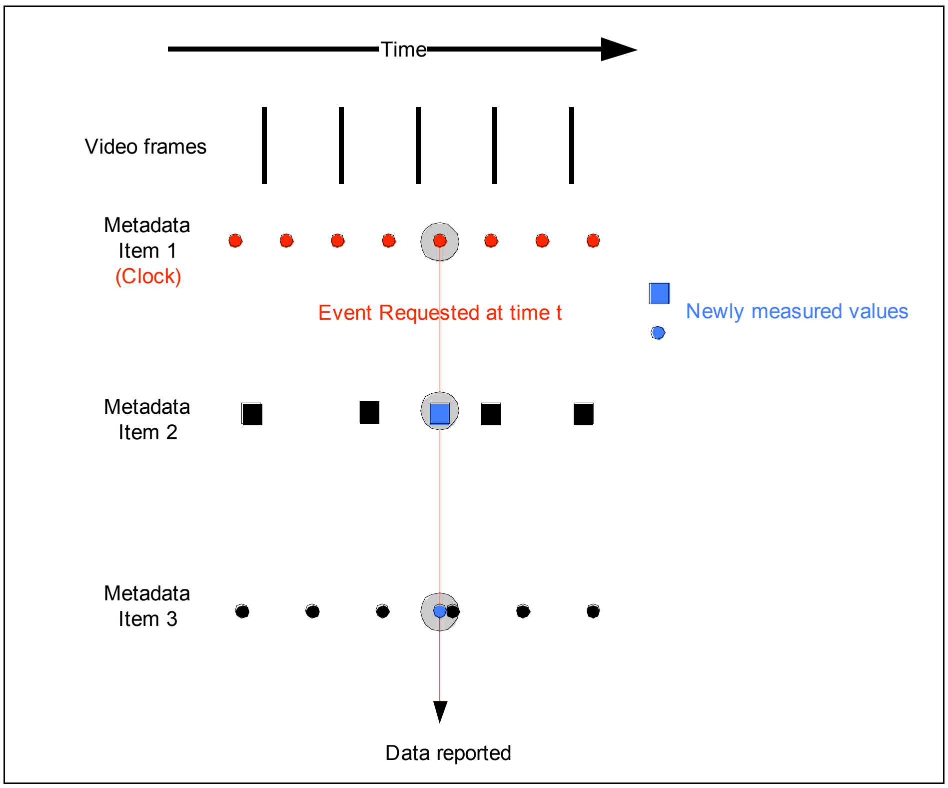


Figure 8, Time Tagged, Event Sampled, Video Asynchronous Metadata

1.6 *Time Tagged, Group Sampled, Video Asynchronous Metadata*

This diagram shows video asynchronous time stamped metadata that is sent from the sensor at a regular interval. An example would be time tagged metadata sent from the sensor at 13Hz and the video collected at 30Hz.

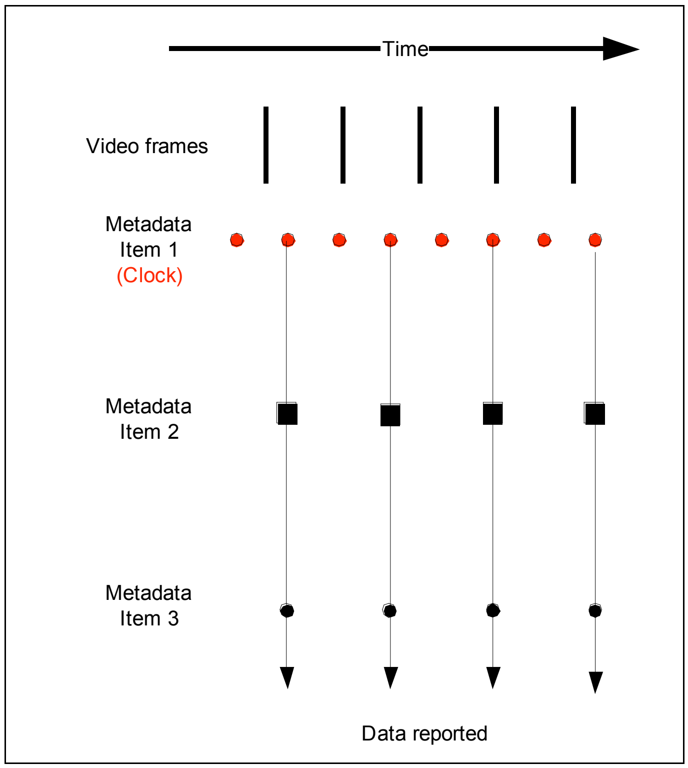


Figure 9, Time Tagged, Group Sampled, Video Asynchronous Metadata

1.7 Time Tagged, Group Sampled Metadata

The following diagrams show time stamped metadata that is sent at regular intervals. The video isochronous diagram shows the metadata is out of phase with the frames but at a multiple of the frame rate. An example would be time tagged metadata sent from the sensor at 15Hz plus a phase shift of 10 milliseconds and the video collected at 30Hz. The video synchronous diagram shows the metadata is in phase with a multiple of the frame rate. An example would be time tagged metadata sent from the sensor at 15Hz and the video collected at 30Hz. The video synchronous at video frame rate diagram shows the metadata is in phase with every frame of the video. If the metadata can be sampled at the frame rate then this is the most accurate method for time tagging metadata.

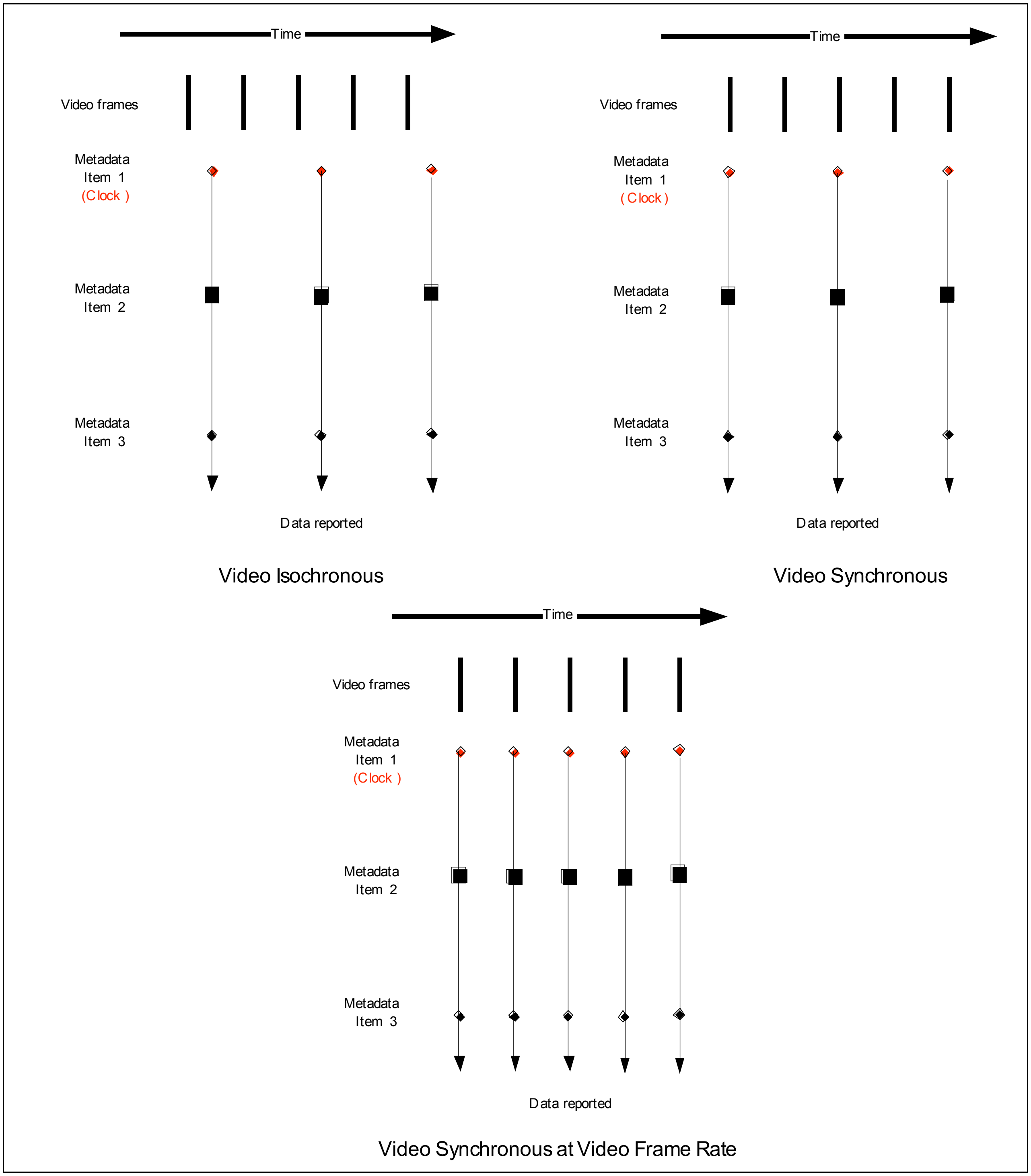


Figure 10, Time Tagged, Group Sampled Metadata

1.8 Time Tagged, Individual Sampled, Video Asynchronous Metadata

The following diagram shows time stamped metadata that is sent at the rate that each metadata item is generated. If the metadata cannot be sampled at the frame rate then this is the most accurate method for time tagging metadata.

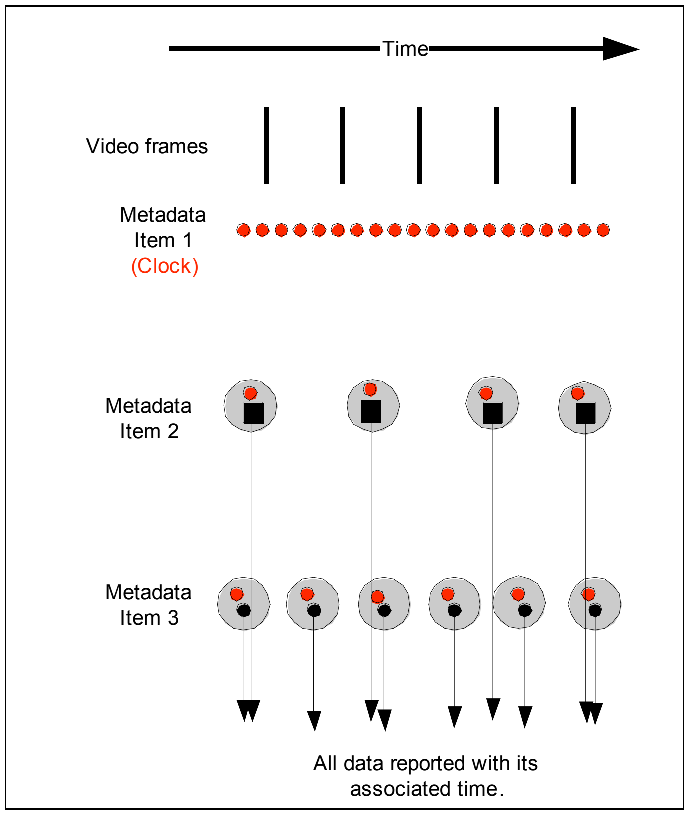


Figure 11, Time Tagged, Individual Sampled, Video Asynchronous Metadata

Annex C Proof-of-Concept Demonstration (Informative)

Global Synchronization and Time Stamping of Digital Video Using GPS

A proof-of-concept demonstration was performed using COTS Broadcast Television equipment that demonstrated the capability to globally synchronize “electronic shutters” and time stamp the video streams utilizing GPS as the common reference (Fig. 2). The GPS clock(s) maintains time accuracy (traceable) to within 1 microsecond of the U.S. Naval Observatory. GPS receivers maintain relative synchronization to each other typically within 15 nanoseconds. Figure 1 is a log from the demonstration receiver while traveling on a roadway. The peaks in error were induced from driving under overpasses and loosing satellite link. GPS receivers for this purpose provide high precision (Stratum 1) 10MHz reference, Time in UTC and 1-PPS (See Fig. 4-5). Master Sync Generators in turn derive their reference sync signals by processing the 1-PPS and 10 MHz from GPS and precisely determining the beginning of each video frame and time code bit (See Fig. 3). This process is called “Genlocking” the reference generator. UTC is converted to SMPTE 12M/309 Time Code for embedding in the video stream using SMPTE RP188. Using this technique any two cameras (sensors), locally or globally, can be synchronized to ~20 nanosecond accuracy or better. (One 30fps video frame is 33ms)

The demonstration showed the coincidence of three critical timing signals;

1. PPS with Start of First Frame of Video (Fig. 6-8)
2. PPS with Start of Time Code, and therefore First Frame Video (Fig. 9-11)
3. UTC for Time Code Value using SMPTE 12M/309 and RP188

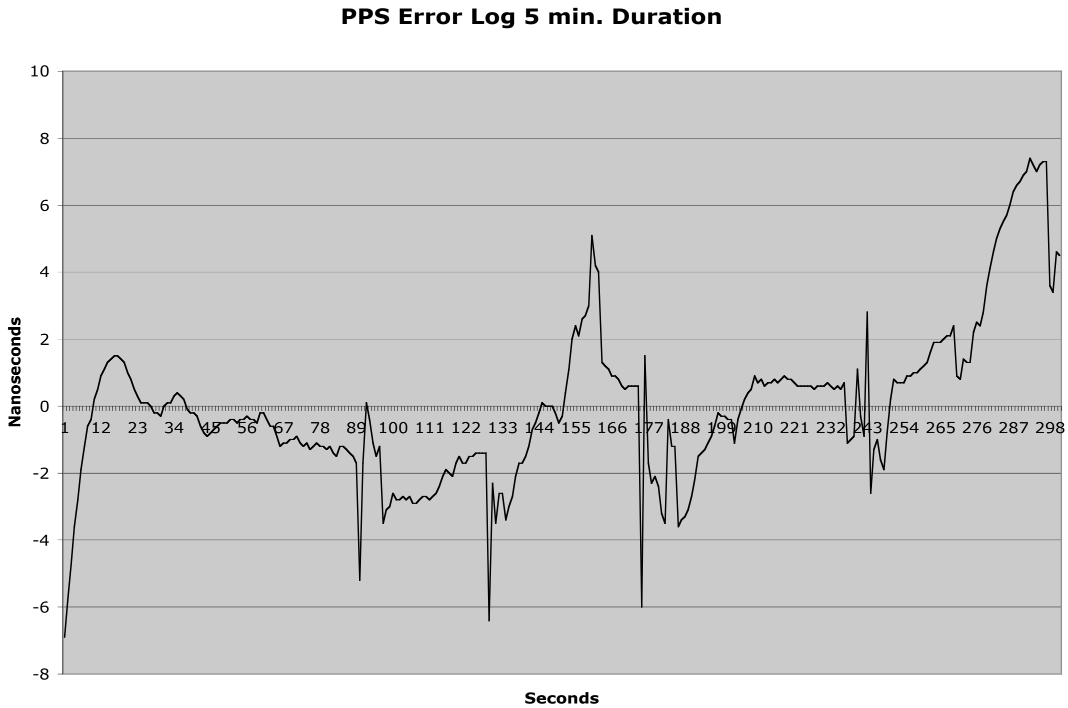


Figure 12: PPS Error Log Receiver in motion

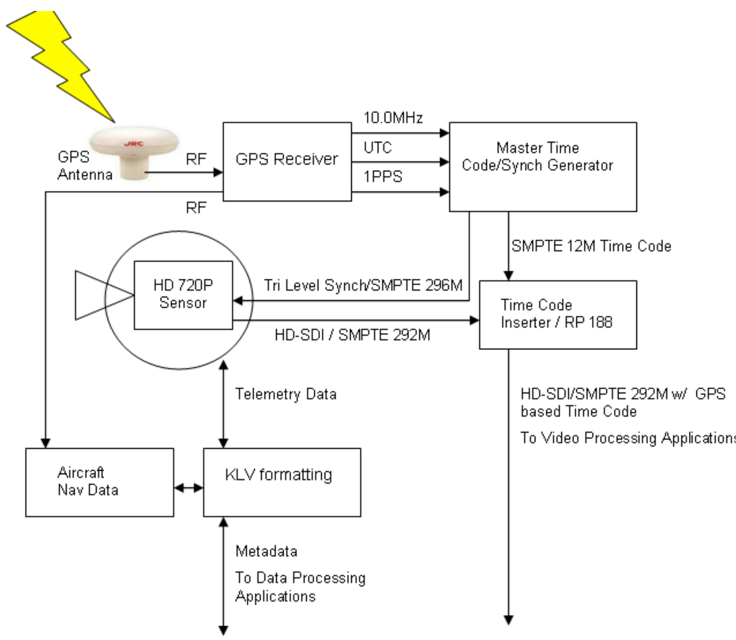
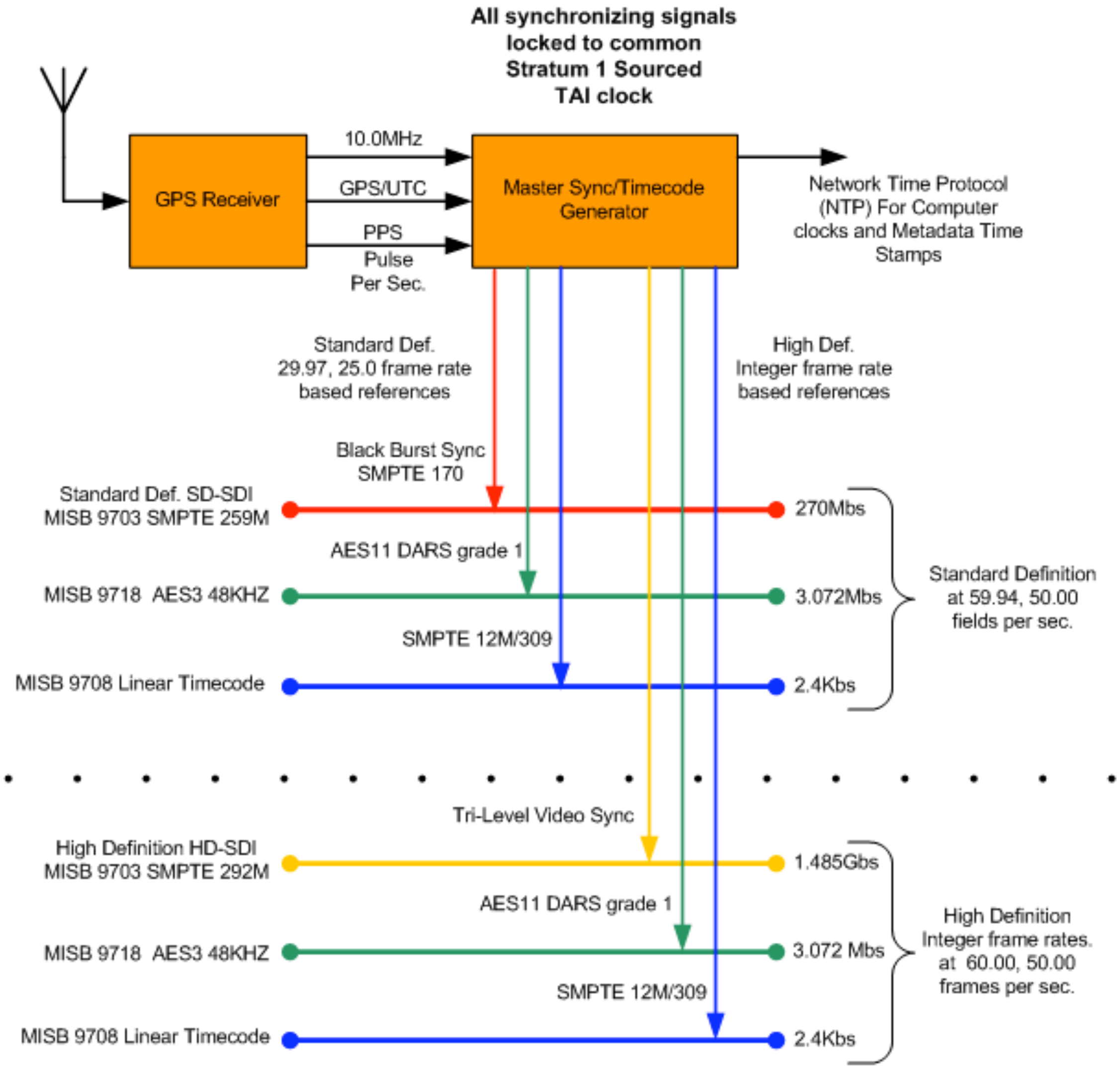


Figure 13 Airborne Reference implementation using HD Video 720x1080p SMPTE 296M



GPS Synchronization

Baseband Transport Streams

Figure 14: GPS Synchronized MISB/SMPTE video, audio, time code signals

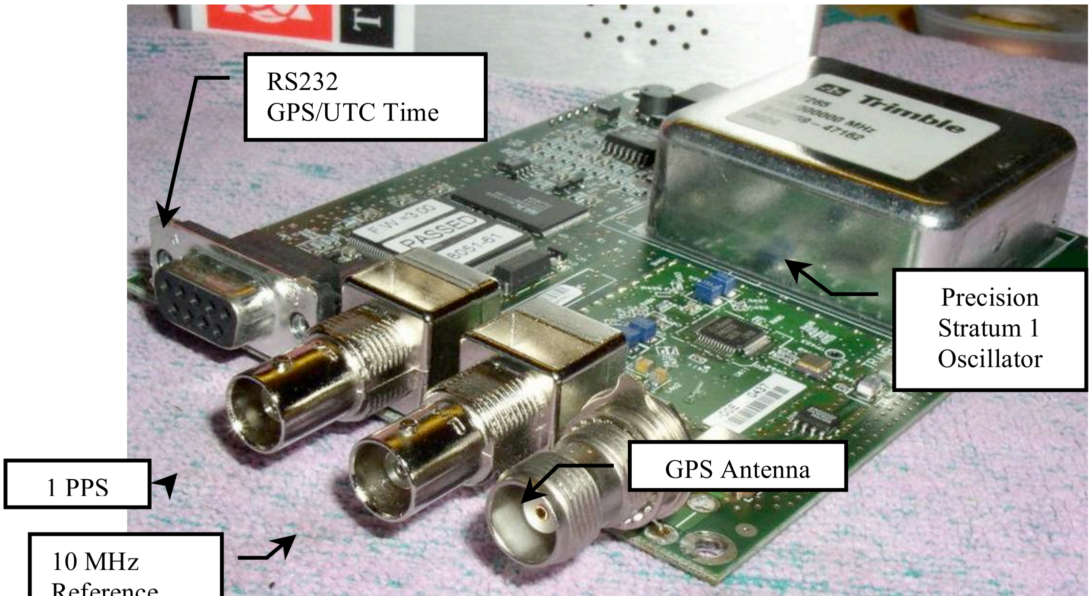


Figure 15: GPS Receiver Un-cased

Figure 16: GPS Receiver connected to Master Video Sync Generator

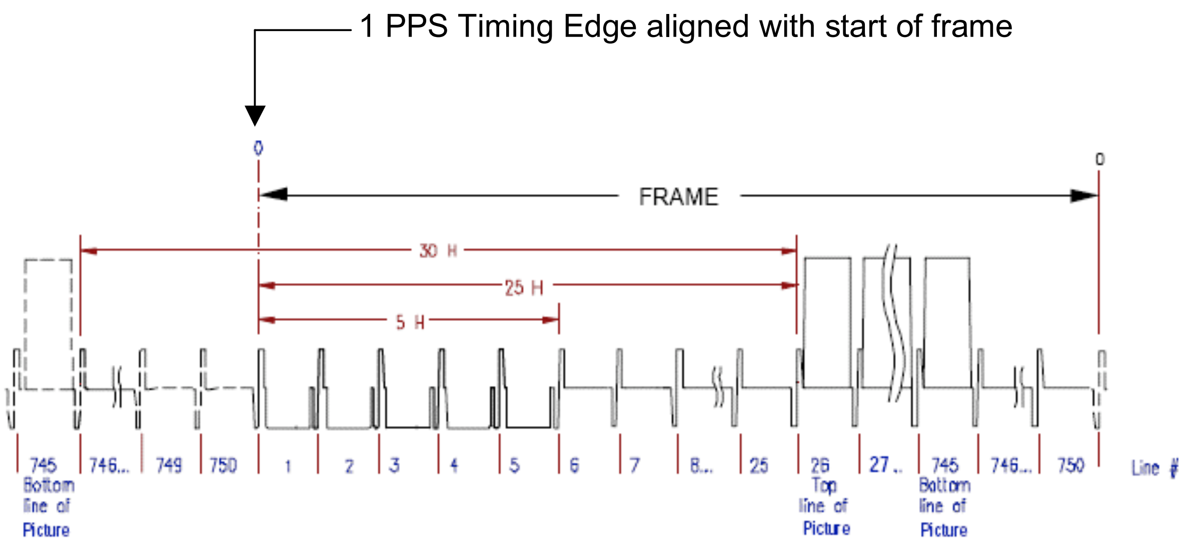


Figure 17: GPS PPS Sync Point in HD 720p-60Fps Tri Level Sync, SMPTE 296M

Scope Photos of Prototype Sync System

The photos below show the timing relationships and alignment between GPS 1PPS, Tri Level Sync, the start of a video frame

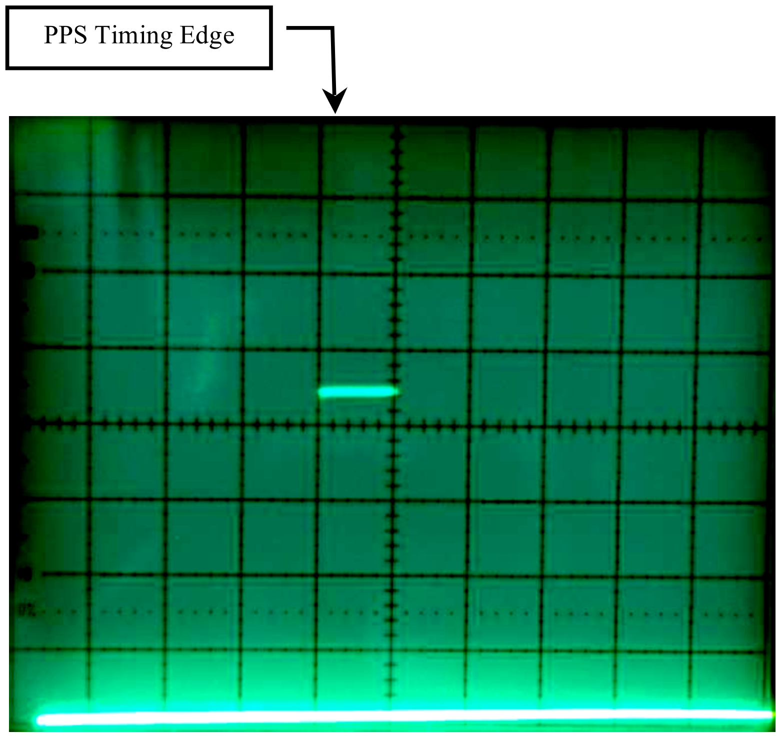


Figure 18: 1-PPS 10 usec/div.

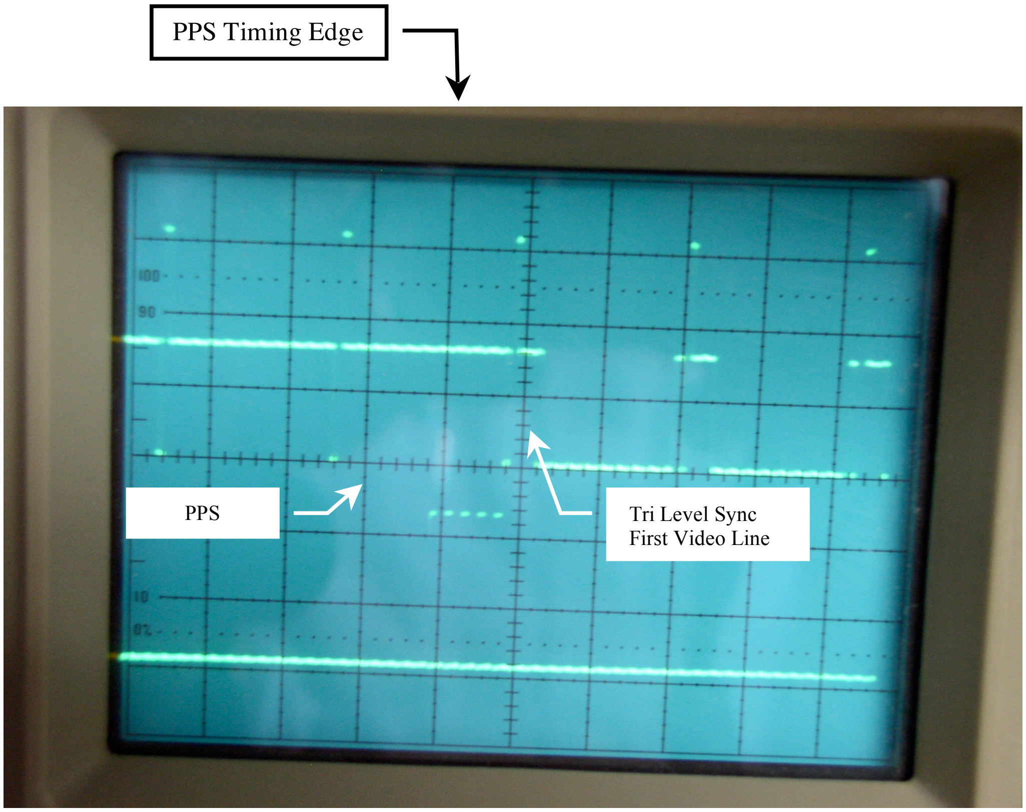


Figure 19: Pulse Per Second aligned with Tri Level Sync Start of Video Frame 10 usec/div.

Scope Photos of Prototype Sync System

The next series of photos show the timing relationships and alignment between GPS 1PPS, Tri Level Sync, the Start of a Video Frame, and Time Code in increasing sweep resolutions.

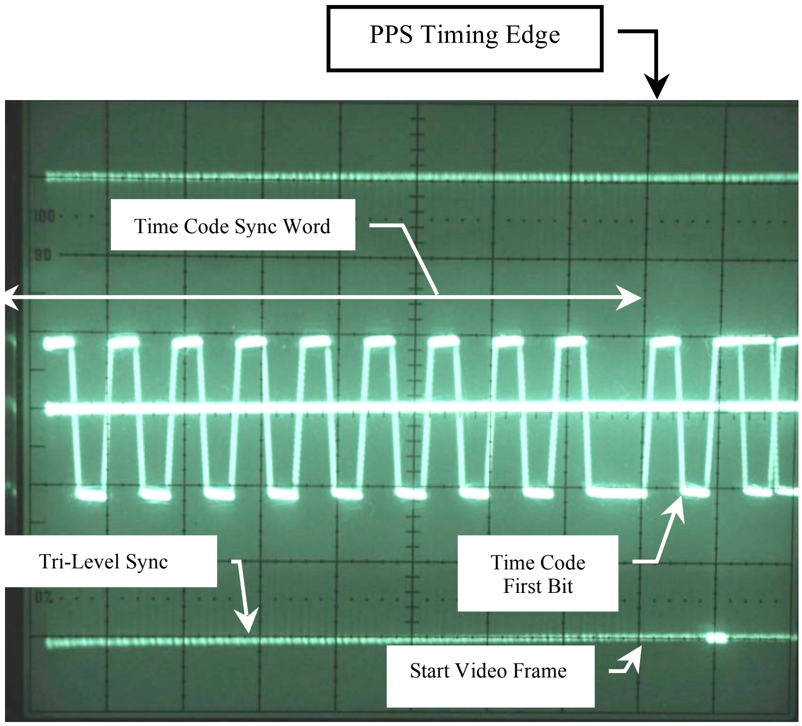


Figure 20:Time Code superimposed on Tri-Level Sync. 0.5ms per division

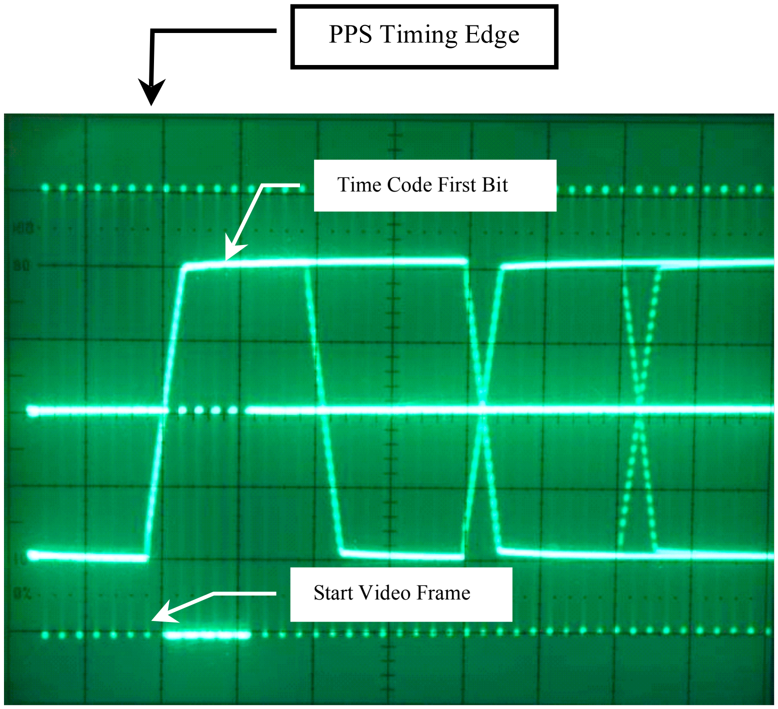


Figure 21: Expanded Scale of Fig.6 showing alignment of Time Code first bit and First Frame of Video 100 usec/div.

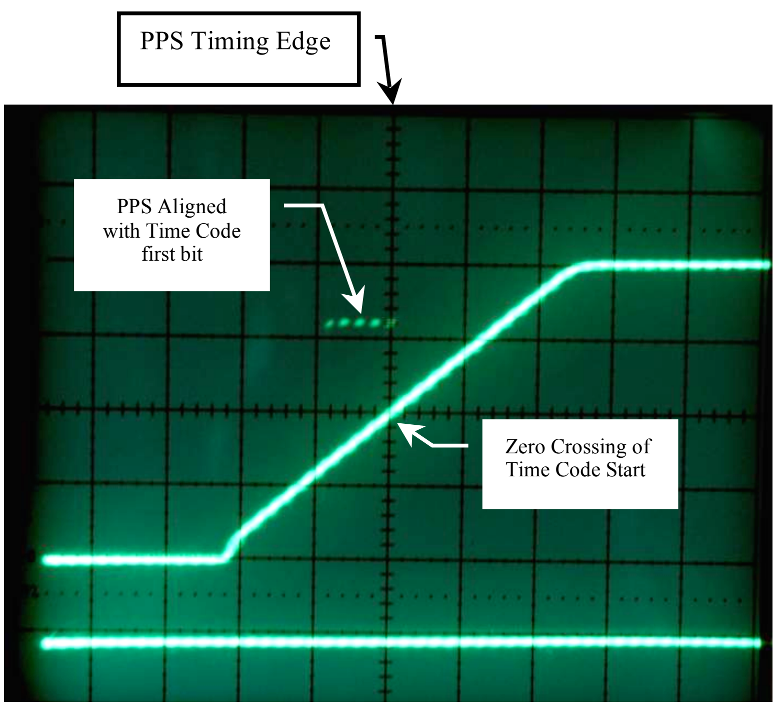


Figure 22: 1-PPS with Time code Start 10 usec/div.

Figure 23: Prototype Assembled Components